

## **What is claimed is:**

**[Claim 1]** A software diagnostics platform comprising:

a command interface adapted to receiving commands and outputting results;  
an engine adapted to running a test sequence;  
an output driver adapted to timestamping an outgoing message and storing said outgoing message;  
an input driver adapted to timestamping an incoming message and storing said incoming message; and  
an analysis routine adapted to analyzing said outgoing message and said incoming message;  
said software system being operable on an embedded processor.

**[Claim 2]** The software diagnostics platform of claim 1 wherein said command interface is operable to communicate via a terminal interface.

**[Claim 3]** The software diagnostics platform of claim 1 further comprising:  
a host program operable on a host system, said host program having a graphical user interface.

**[Claim 4]** The software diagnostics platform of claim 1 wherein said test sequence comprises a single test routine.

**[Claim 5]** The software diagnostics platform of claim 1 wherein said test sequence comprises a test routine that is repeated for a predetermined period of time.

**[Claim 6]** The software diagnostics platform of claim 1 wherein said test sequence comprises multiple threads of commands.

**[Claim 7]** The software diagnostics platform of claim 1 wherein input driver is further adapted to validate said incoming message.

**[Claim 8]** The software diagnostics platform of claim 1 wherein said analysis comprises determining one of a group comprising message transfer time, average message transfer time, and average data throughput per unit time.

**[Claim 9]** The software diagnostics platform of claim 1 further comprising an initiator adapted to determine if an I/O device is present.

**[Claim 10]** The software diagnostics platform of claim 9 wherein said initiator is further adapted to perform a diagnostic routine with said I/O device.

**[Claim 11]** A system comprising:

a device with an embedded processor, said device having a specific function;  
a first software system operable to run on said embedded processor and enable said device to perform said specific function; and  
a second software system operable to run on said embedded processor, said second software system comprising:  
a command interface adapted to receiving commands and outputting results;  
an engine adapted to running a test sequence;  
an output driver adapted to timestamping an outgoing message and storing said outgoing message;  
an input driver adapted to timestamping an incoming message and storing said incoming message; and  
an analysis routine adapted to analyzing said outgoing message and said incoming message.

**[Claim 12]** The system of claim 11 wherein said command interface is operable to communicate via a terminal interface.

**[Claim 13]** The system of claim 11 further comprising:

a host program operable on a host system, said host program having a graphical user interface.

**[Claim 14]** The system of claim 11 wherein said test sequence comprises a single test routine.

**[Claim 15]** The system of claim 11 wherein said test sequence comprises a test routine that is repeated for a predetermined period of time.

**[Claim 16]** The system of claim 11 wherein said test sequence comprises multiple threads of commands.

**[Claim 17]** The system of claim 11 wherein input driver is further adapted to validate said incoming message.

**[Claim 18]** The system of claim 11 wherein said analysis comprises determining one of a group comprising message transfer time, average message transfer time, and average data throughput per unit time.

**[Claim 19]** The system of claim 11 further comprising an initiator adapted to determine if an I/O device is present.

**[Claim 20]** The system of claim 19 wherein said initiator is further adapted to perform a diagnostic routine with said I/O device.

**[Claim 21]** A test system comprising:

- a reusable test sequence;

- a first command interpreter adapted to interpret said reusable test sequence, said first command interpreter being adapted to operate on a first embedded processor; and

- a second command interpreter adapted to interpret said reusable test sequence, said second command interpreter being adapted to operate on a second embedded processor;

wherein said first command interpreter and said second command interpreter each comprise:

- a command interface adapted to receiving commands and outputting results;

- an engine adapted to running a test sequence;

- an output driver adapted to timestamping an outgoing message and storing said outgoing message;

- an input driver adapted to timestamping an incoming message and storing said incoming message; and

- an analysis routine adapted to analyzing said outgoing message and said incoming message.

**[Claim 22]** A method of developing a circuit having an embedded processor comprising:

- designing a circuit having said embedded processor, said circuit having a predefined function;

assembling said circuit;  
designing software operable on said embedded processor, said software adapted to enable said circuit to perform said predefined function;  
loading said embedded processor with a test platform software comprising:  
a command interface adapted to receiving commands and outputting results;  
an engine adapted to running a test sequence;  
an output driver adapted to timestamping an outgoing message and storing said outgoing message;  
an input driver adapted to timestamping an incoming message and storing said incoming message;  
an analysis routine adapted to analyzing said outgoing message and said incoming message and create results; and  
a display routine for displaying said results;  
creating said test sequence;  
transmitting said test sequence to said embedded processor loaded with said test platform software;  
operating said test sequence on said embedded processor; and  
analyzing said results.